#30 Attach

PATENT

Attorney Docket No. 11823-002630

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Examiner: J. Burke
CARY L. QUEEN ET AL.	Art Unit: 1642
Application No.: 08/484,537	
Filed: June 7, 1995	
For: IMPROVED HUMANIZED IMMUNOGLOBULINS	

DECLARATION OF MAXIMILIANO VASQUEZ

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

I, Maximiliano Vasquez, declare and state as follows:

1. I received my Ph.D. in 1987 from Cornell University (Ithaca, NY). I am an author of over 30 scientific publications, many of which report on research in protein structure, including antibody structure and humanization. I am now a Senior Scientist at Protein Design Labs, Inc. In this capacity, one of my primary responsibilities is to participate in the design of the company's humanized antibodies. A copy of my curriculum vitae is attached as Exhibit 1.

2. I have reviewed the subject Patent Application, the Office Action dated April 29, 1999, and the references George et al. and Barton et al. cited therein.

3. I understand that the Examiner takes the position that the specification has not enabled determining which sequences are 65% or 70% identical, because sequence identity has no common meaning within the art, since the scoring of gaps when comparing one sequence to another introduces uncertainty as to the percent of similarity. Although this may be correct with respect to certain protein sequences, it is not correct with respect to immunoglobulin (Ig) heavy chain variable region framework sequences, which are compared in the claims, for the reasons stated below.

4. I conducted a study to determine whether the scoring of gaps would in fact affect the alignment of Ig heavy chain framework sequences and thus the percent identity. I used as an example the heavy chain framework sequences of the mouse anti-Tac antibody and the human Eu antibody, because these provided the first experimental example in the Application.[1] However, I believe I would have obtained similar results with any other heavy chain framework sequences.

5. To align these framework sequences, I used the GAP program of the Wisconsin Package for sequence analysis. This software package, which was developed by the Genetics Computer Group (Madison, WI) is widely used in the scientific community. Moreover, the GAP program offers a full range of algorithms to align two sequences, because the gap penalty (both gap creation penalty and gap extension penalty) as well as the amino acid similarity matrix may be chosen by the user. The chapter of the user manual describing the GAP program is attached to this Declaration as Exhibit 2. Gap penalities and similarity matrices are described at length in that chapter as well as by George et al. and Barton et al.

6. Initially, I used three similarity matrices -- BLOSUM62, PAM250 and the Identity Matrix -- because these are particularly preferred by scientists performing sequence alignment (see Barton et al., p. 31-32 and p. 34-35). For each matrix, I first used the default values for the gap creation penalty and gap extension penalty provided by the program, because these have been chosen to work especially well with the respective matrices. In addition, I then performed another alignment using each matrix, but with alternative gap penalties that I chose, so that they were either more or less stringent than the default gap penalties.

7. The exact outputs produced by the GAP program for these 6 alignments - using the 3 matrices, each with the default and alternative gap penalties - are attached as Exhibit 3. Each output lists the sequences being aligned (mouse anti-Tac and human Eu heavy chain frameworks), the similarity matrix and gap penalties being used (denoting the gap creation penalty as "gap weight" and the gap extension penalty as "length weight"), the alignment itself, and the percent identity derived from the alignment. The definition of percent identity used by the program agrees with that commonly understood by scientists: "Percent Identity is the percent of symbols that actually match" (see the fourth line of page G-6 of Exhibit 2).

8. Inspection of the outputs in Exhibit 3 shows immediately that all the algorithms (i.e., different matrices and gap penalties) produced precisely the same alignment and percent identity (58 of 87 matches, or 66.667%). To verify that the same results would also be produced using less well-known similarity matrices and still other gap penalities, I used the program with 4 other matrices and the default gap penalties provided by the program: BLOSUM30, gap creation = 15, gap extension = 5; BLOSUM100, gap creation = 19, gap extension = 10; PEP matrix, gap creation = 30, gap extension = 1; STRUCTGAPPEP matrix, gap creation = 40, gap extension = 5. Indeed, as predicted, each of these algorithms generated precisely the same alignment and percent identity (66.667%) as the 6 algorithms described above.

9. I also verified directly that the alignment produced by all these algorithms was the same as the alignment generated by Kabat numbering.[2] In particular, the alignment did not contain gaps in either sequence (although the algorithms certainly would have allowed gaps if that had given the optimal alignment, taking into account the gap penalties). This was in accord with the general scientific understanding that Ig framework sequences almost never have gaps when aligned.

10. The matrices and gap penalities I used were chosen to cover a wide range of biologically reasonable possibilties, but of course the analysis cannot include all the infinite number of possible gap penalties. Hence, it is quite possible that some selection of gap penalties, especially if unsuitable or unreasonable, would give a different alignment. However, I do not believe that this would in any way hamper the ordinary skilled scientist from arriving at the same answer for percent identity, because any reasonable algorithm gave the same result (66.667%).

11. Finally, I also want to remark that it is well-known by experts in antibody structure that alignment by Kabat numbering corresponds to the closest physical juxtaposition of the 3-D structures of the frameworks of two immunoglobulin molecules. Hence, even if an unusual choice of gap penalties resulted in some other alignment, scientists familiar with antibody structure would reject it as not being biologically relevant.

12. In conclusion, I have shown by actual test that a wide range of algorithms with various gap penalties all produce the same alignment, as well as percent identity, of two Ig heavy chain framework sequences, and that is the same alignment given by Kabat numbering. Hence,

regarding such framework sequences, the Office Action is not correct that scoring of gaps introduces uncertainty or that percent identity does not have a common meaning in the art.

I further declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statments and the like so made are punishable by fine or imprisonment, or both, under section 1001 of Title 18 of the United States Code, and that such willful false statments may jeopardize the validity of the application or any patent issuing thereon.

Respectfully submitted,

Dated: July 12, 1999

---

[1] To perform the alignment, I input only the framework sequences for the anti-Tac and Eu heavy chain variable regions, omitting the Kabat CDRs. These framework sequences, which each have 87 amino acids, are shown aligned in Exhibit 3 described below.

[2] The actual numbers output by the program in Exhibit 3 are sequential numbers and not Kabat numbers, since the GAP program is not specific for Ig sequences but can align any protein sequences. However, the alignment itself is the same as that produced by Kabat numbering.

Exhibit 1

Maximiliano Vásquez
34801 Campus Drive
Fremont, CA 94555
(510) 574 1477

PROFESSIONAL EMPLOYMENT

March, 1998 to present (Senior)

May, 1990 to February 1998 (Staff)

STAFF AND SENIOR SCIENTIST, PROTEIN DESIGN LABS, INC. 34801 CAMPUS DRIVE, FREMONT, CA 94555 I have worked on development of computational tools for modeling and analysis of antibody structure, and for analysis of the large amount of sequence data available for antibody variable domains. With the help of this program, more than twenty mouse antibodies have been successfully humanized. I have also developed a series of modules for drug design projects. I devised an improved new procedure to compute side-chain conformations in globular proteins.

September, 1988 to May, 1990

SENIOR APPLICATIONS SCIENTIST, TRIPOS ASSOCIATES, INC. 1699 S. HANLEY RD. ST. LOUIS, MO 63144. My main project was integration into Sybyl of Composer, a collection of programs for protein modeling by homology. In addition, I provided general scientific direction to the software engineering group involved in development of the Sybyl/Biopolymer module.
I worked on the specification, design, testing, and validation of the Molecular Dynamics module of Sybyl, which was first released in early 1989. I consulted with a number of Tripos users in industry and academia on applications of the Sybyl molecular modeling system, including protein and small molecule modeling, active analog approach, and QSAR.

August, 1987 to August, 1988

POSTDOCTORAL RESEARCH ASSOCIATE, CORNELL UNIVERSITY BAKER LABORATORY OF CHEMISTRY ITHACA, NY 14853. This research was carried out in Professor Scheraga's laboratory. It included a theoretical investigation of methods for the consideration of the effect of hydration on the conformations of polypeptides and proteins. We applied these, as well as chain build-up and Monte Carlo techniques, to calculate stable structures of a small cyclic peptide. I also extended some of my early distance geometry work to deal with actual NMR data obtained for a peptide-enzyme complex, and produce structures of the peptide in the bound state using transfer NOE data.

January 1980 to December 1980

TEACHING INSTRUCTOR, PHYSICAL CHEMISTRY LABORATORY, UNIVERSIDAD DE COSTA RICA. I taught a course of experimental physical chemistry to junior-level Chemical Engineering students.

EDUCATION

1983 to 1987

Cornell University, Ithaca, New York. Ph.D., Biophysical Chemistry

Graduate Research Work

It was conducted in the laboratory of Harold A. Scheraga. My Ph.D. research involved use of conformational energy and distance geometry calculations to obtain protein structures consistent with simulated nuclear magnetic resonance data. The simulated data were derived from known three-dimensional structures determined by X-ray diffraction. We applied these techniques to rebuild the structures of the proteins crambin and pancreatic trypsin inhibitor (pti) from limited distance information.

I was involved in other research projects not directly related to my doctorate thesis. In collaboration with Matthew Pincus, then at the Department of Pathology of the New York University Medical Center, we applied one-dimensional physical models to explore a hypothetical correlation between the α-helical tendency and the biological activity of a series of polypeptide molecules in a T-lymphocyte proliferation assay.

I worked in collaboration with Hagai Meirovitch, then at the Polymer Research Department of the Weizmann Institute in Israel, to adapt some of his ideas for calculation of the free energy of very simplified and abstract polymer models, to more realistic, atomic level, models of polypeptides.

1981 to 1983	Cornell University, Ithaca, New York. M. Sc., Biophysical Chemistry
1976 to 1979	Universidad de Costa Rica, San Jose, Costa Rica - Central America. B.Sc., Chemistry

PUBLICATIONS

1. M. Vásquez, G. Némethy & H. A. Scheraga (1983) 'Computed Conformational States of the 20 Naturally Occurring Amino Acids and of the Prototype Residue α-Amino Butyric Acid' *Macromolecules* **16**, 1043-1049.
2. M. Vásquez & H.A. Scheraga (1985) 'Use of Buildup and Energy-Minimization Procedures to Compute Low-Energy Structures of the Backbone of Enkephalin' *Biopolymers* **24**, 1437-1447.
3. M. Vásquez, M.R. Pincus & H.A. Scheraga (1987) 'Helix-Coil Transition Theory Including Long-Range Electrostatic Interactions: Application to Globular Proteins' *Biopolymers* **26**, 351-371.
4. M. Vásquez, M.R. Pincus & H.A. Scheraga (1987) 'Correlation Between Computed Conformational Properties of Cytochrome c Peptides and their Antigenicity in a T-Lymphocyte Proliferation Assay' *Biopolymers* **26**, 373-386.
5. H. Meirovitch, M. Vásquez & H.A. Scheraga (1987) 'Stability of Polypeptide Conformational States as Determined by Computer Simulation of the Free Energy' *Biopolymers* **26**, 651-671.
6. M. Vásquez & H.A. Scheraga (1988) 'Effect of Sequence-Specific Interactions on the Stability of Helical Conformations in Polypeptides' *Biopolymers* **27**, 41-58.
7. M. Vásquez & H.A. Scheraga (1988) 'Calculation of Protein Conformation by the Buildup Procedure. Application to Bovine Pancreatic Trypsin Inhibitor Using Limited Simulated Nuclear Magnetic Resonance Data' *J. Biomol. Struct. Dynamics* **5**, 705-755.
8. M. Vásquez & H.A. Scheraga (1988) 'Variable-Target Function and Buildup Procedures for the Calculation of Protein Conformation. Application to Bovine Pancreatic Trypsin Inhibitor Using Limited Simulated Nuclear Magnetic Resonance Data' *J. Biomol. Struct. Dynamics* **5**, 757-784.
9. H. Meirovitch, M. Vásquez & H.A. Scheraga (1988) 'Stability of Polypeptide Conformational States: II Folding of a Polypeptide Chain by the Scanning Simulation Method, and Calculation of the Free Energy of the Statistical Coil' *Biopolymers* **27**, 1189-1204.

10. F. Ni, Y.C. Meinwald, M. Vásquez & H.A. Scheraga (1989) 'High-Resolution NMR Studies of Fibrinogen-like Peptides in Solution: Structure of a Thrombin-bound Peptide Corresponding to Residues 7-16 of the A-α Chain of Human Fibrinogen' *Biochemistry* **28**, 3094-3105.
11. H. Meirovitch, M. Vásquez & H.A. Scheraga (1990) 'Stability of Polypeptide Conformational States: III The Double Scanning Simulation Method for Calculation of the Free Energy of Polypeptide Chain' *J. Chem. Phys.* **92**, 1248-1257.
12. K.H. Altman, J. Wocjik, M. Vásquez & H.A. Scheraga (1990) 'Helix-Coil Stability Constants for the Naturally Occurring Amino Acids in Water. 23. Characterization of Proline from Random Poly (Hydroxybutylglutamine-co-L-Proline)' *Biopolymers* **30**, 107-120.
13. D.R. Ripoll, L. Piela, M. Vásquez & H.A. Scheraga (1991) 'On the Multiple-Minima Problem in the Conformational Analysis of Polypeptides. V. Application of the Self-Consistent Electrostatic Field and the Electrostatically-Driven Monte Carlo Methods to Bovine Pancreatic Trypsin Inhibitor' *Proteins* **10**, 188-198.
14. J. Vila, R.L. Williams, M. Vásquez & H.A. Scheraga (1991) 'Empirical Solvation Models Can Be Used to Differentiate Native From Near-Native Conformations of Bovine Pancreatic Trypsin Inhibitor' *Proteins* **10**, 199-218.
15. D.R.. Ripoll, M. Vásquez & H.A. Scheraga (1991) 'The Electrostatically-Driven Monte Carlo Method: Application to Conformational Analysis of Decaglycine' *Biopolymers* **31**, 319-330.
16. F.L. Sebastiani, L.B. Farrell, M. Vásquez & R.N. Beachy (1991) 'Conserved Amino Acid Sequences Among Plant Proteins Sorted to Protein Bodies and Plant Vacuoles. Can They Play a Role in Protein Sorting?' *Eur.J.Biochem.* **199**, 441-450.
17. S.M. Glaser, M. Vásquez, P.W. Payne & W.P. Schneider (1992) 'Dissection of the Combining Site in a Humanized Anti-Tac Antibody' *J. Immunology* **149**, 2607-2614.
18. J.A. Simpson, J.C. Chow, J. Baker, N. Avdalovic, S. Yuan, D. Au, M.S. Co, M. Vásquez, W.J. Britt & K.L. Coelingh (1993) 'Neutralizing Monoclonal Antibodies That Distinguish Three Antigenic Sites on Human Cytomegalovirus Glycoprotein H Have Conformationally Distinct Binding Sites' *J. Virology* **67**, 489-496.
19. M.S. Co, D.A. Scheinberg, N.M. Avdalovic, K. McGraw, M. Vásquez, P.C. Caron & C. Queen (1993) 'Increasing the Affinity of an anti-CD33 Monoclonal Antibody by Genetically Engineered Deglycosylation of the Variable Domain' *Molecular Immunology* **30**, 1361-1367.

20. M. Vásquez & P.W. Payne (1993) 'Computational Approaches to the Design of Therapeutic Antibodies with Enhanced Clinical Efficacy' *Chem. Design. Autom. News.* **8**, 16-25.
21. 7. M.S. Co, S. Yano, R.K. Hsu, N.F. Landolfi, M. Vásquez, M. Cole. J.T. Tso, T. Bringman, W. Laird, D. Hudson, K. Kawamura, K. Suzuki, K. Furuichi, C. Queen & Y. Masuho (1994) 'A Humanized Antibody Specific for the Platelet Integrin gpIIb/IIIa' *J. Immunology* **152**, 2968-2976.
22. H. Meirovitch, E. Meirovitch, A.G. Michel & M. Vásquez (1994) 'A Simple and Effective Procedure for Conformational Search of Macromolecules: Application to Met- and Leu-Enkephalin' *J. Phys. Chem.* **98**, 6241-6243.
23. M. Vásquez, G. Némethy & H. A. Scheraga (1994) 'Conformational Energy Calculations on Polypeptides and Proteins' *Chemical Reviews* **94**, 2183-2239.
24. M. Vásquez, E. Meirovitch & H. Meirovitch (1994) 'A Free Energy Based Monte Carlo Minimization Procedure for Biomolecules' *J. Phys. Chem.* **98**, 9380-9382.
25. M. Vásquez (1995) 'An Evaluation of Discrete and Continuum Search Techniques for Conformational Analysis of Side Chains in Proteins' *Biopolymers*, **36**, 53-69.
26. S. Kumar, P.W. Payne & M. Vásquez (1996) 'Free-Energy Calculations Using Iterative Techniques' *J. Comp. Chem.* **17**, 1269-1275.
27. Z. Zhou, N. Kuhn, P. Payne, M. Vásquez & M. Levitt (1996) 'Finite Difference Solution of the Poisson-Boltzmann Equation: Complete Elimination of Self-Energy' *J. Comp. Chem.* **17**, 1344-1351.
28. M. Vásquez (1996) 'Modeling Side Chain Conformation' *Current Opinion Struct. Biol.* **6**, 217-221.
29. M.S. Co, J. Baker, K. Bednarik, E. Janzek, W. Neruda, P. Mayer, R. Plot, B. Stumper, M. Vásquez, C. Queen & H. Loibner (1996) 'Humanized Anti-Lewis Y Antibodies: in vitro Properties and Pharmacokinetics in Rhesus Monkeys' *Cancer Research* **56**, 1118-1125.
30. H. Meirovitch & M. Vásquez (1997) 'Efficiency of simulated annealing and the Monte Carlo minimization method for generating a set of low energy structures of peptides' J. Mol. Struct: THEOCHEM **398**, 517-521.
31. X.-Y. He, Z. Xu, J. Melrose, A. Mullowney, M. Vásquez, C. Queen, V. Vexler, C. Klingbeil, M.S. Co & E. L. Berg (1998) 'Humanization and Pharmacokinetics of a Mouse Monoclonal Antibody with Specificity for Both E- and P-Selectin' *J. Immunology* **160**, 1029-1035.
32. M.S. Co, N.F. Landolfi, J.O. Nagy , J.H. Tan, V. Vexler, M. Vásquez, L. Roark, S.Yuan, P.R. Hinton, J. Melrose, C. Klingbeil, C. Queen & E.L. Berg (1999)

'Properties and pharmacokinetics of two humanized antibodies specific for L-selectin' *Immunotechnology* **4**, 253-266.

33. Z.C. Fan, L. Shan, B.Z. Goldsteen, L.W. Guddat, A. Thakur, N.F. Landolfi, M.S. Co, M. Vásquez, C. Queen, P.A. Ramsland & A.B. Edmundson (1999) 'Comparison of the Three-Dimensional Structures of a Humanized and a Chimeric Fab of an anti-γ-interferon Antibody' *J. Molecular Recognition* **12**, 19-32.

Exhibit 2

Wisconsin Package

Program Manual

Version 9

UNIX

Genetics Computer Group
University Research Park
575 Science Drive
Madison, WI 53711

Phone: (608) 231–5200
Fax: (608) 231–5202
E–mail: help@GCG.com
WWW: http://www.gcg.com/

A wholly owned subsidiary of **Oxford Molecular Group, Inc.**

Cover Photo

The cover photo is a "fragment assembly" of the garden pea *Pisum sativum* with the "A" allele, which confers red flowers. This plant is a member of the Marx genetic stock collection of the USDA Plant Germplasm System, a legacy of the USA's premier pea geneticist, the late Dr. Gerald A. Marx, formerly of Cornell University. We thank Dr. Chuck Simon and Dr. Richard Hannon of the USDA, ARS, NPGS, Regional Plant Introduction Station at Washington State University, Pullman, WA, USA for aiding us in acquiring these photos.

Credits

Technical editors: Irv Edelman and Sue Olson.

Cover and tab graphic design: Design Foundry

Production: Joleen Rau, Ann Kiefer, Dawn Stencil, and Bradley Babler

Acknowledgements

Excerpt from "Little Gidding" in *FOUR QUARTETS*, Copyright 1943 by T.S. Eliot and renewed 1971 by Esme Valier Elot, reprinted by permission of Harcourt Brace & Company.

"The Science of Life" originally appeared in the Fall 1995 Issue of the *Wisconsin Academy Review* and is reprinted with permission of the Wisconsin Academy of Sciences, Arts, and Letters and author Robin Chapman.

Document Revision History

Version 9.1, September 1997
Version 9.0, December 1996
Version 8, September 1994
Version 7, April 1991
Version 6, February 1989
Version 5, June 1987
Version 4, April 1986
Version 3, July 1985
Version 2, June 1984
Version 1, Decembler 1983

Printed in the United States of America

GAP

FUNCTION

Gap uses the algorithm of Needleman and Wunsch to find the alignment of two complete sequences that maximizes the number of matches and minimizes the number of gaps.

DESCRIPTION

Gap considers all possible alignments and gap positions and creates the alignment with the largest number of matched bases and the fewest gaps. You provide a *gap creation penalty* and a *gap extension penalty* in units of matched bases. In other words, Gap must make a profit of *gap creation penalty* number of matches for each gap it inserts. If you choose a gap extension penalty greater than zero, Gap must, in addition, make a profit for each gap inserted of the length of the gap times the gap extension penalty. Gap uses the alignment method of Needleman and Wunsch (J. Mol. Biol. **48**; 443-453 (1970)) that has been shown to be equivalent to Sellers (see the CONSIDERATIONS topic below).

EXAMPLE

Two very long operons of haptoglobin genes are aligned with Gap. The alignment from this example is displayed graphically in the example for the GapShow program. The same sequences are compared in the figures included with DotPlot.

```
% gap

 GAP of what sequence 1 ?  hpr.seq

                   Begin (* 1 *) ?
                 End (*  2966 *) ?
                Reverse (* No *) ?

 to what sequence 2 (* hpr.seq *) ?  hpf.seq

                   Begin (* 1 *) ?
                 End (*  2740 *) ?
                Reverse (* No *) ?

 What is the gap creation penalty (* 50 *) ?

 What is the gap extension penalty (* 3 *) ?

 What should I call the paired output display file (* hpr.pair *) ?

 Aligning ..................................................
          ..................................................
          ....................................-.
 Aligning ..................................................
          ..................................................
          ....................................-.............
          .
```

```
          Gaps:    13
       Quality: 24426
 Quality Ratio: 8.915
  % Similarity: 94.897
        Length:  2982

%
```

OUTPUT

Here is the output from this session:

```
 GAP of: hpr.seq  check: 8102  from: 1  to: 2966

Haptoglobin related sequence
HindIII fragment sequenced 12/27/83
  (partially from hpf sequence)

 to: hpf.seq  check: 2624  from: 1  to: 2740

Haptoglobin alpha2
HindIII fragment , region equivalent to hp1f

 Symbol comparison table: /package/share/9.0/gcgcore/data/rundata/nwsgapdna.cmp
 CompCheck: 8760

        Gap Weight:     50      Average Match: 10.000
     Length Weight:      3   Average Mismatch:  0.000

           Quality:  24426             Length:   2982
             Ratio:  8.915               Gaps:     13
Percent Similarity: 94.897   Percent Identity: 94.897

       Match display thresholds for the alignment(s):
                   | = IDENTITY
                   : =   5
                   . =   1

hpr.seq x hpf.seq         September 19, 1996 10:32  ..

                .         .         .         .         .
      1 AAGCTTGGTATGCTCAGAAGCAGCTAAAGCGTGTATGTGGGGCGGAGGGT 50
        |||||||||||||||||||| ||||||| ||||||| |  |    | ||
      1 AAGCTTGGTATGCTCAGAAGCTGCTAAAGTGTGTATGGGCAG....GTGT 46

   //////////////////////////////////////////////////////////////
                .         .         .         .         .
   1749 TTCCTCTTTCTTCAGAGATGATGAATTATTGTAGCTCCTAGCCCTTTCTT 1798
        ||| |||||||| |||||  ||||||||||||
   1678 TTCATCTTTCTTTAGAGAGAATGAATTATTGTA..................  1710
                                   .
                                   .
                                   .
```

```
                .         .         .         .         .
1949 TGGCCCCTAGCCCTTTCAATGAATTTCAGGGAATTGTGAAAATTCCTTTG 1998
       |||||||||||||||||||||||||||||||||||| ||||||||||
1711 ..GCCCCTAGCCCTTTCAATGAATTTCAGGGAATTGTGGAAATTCCTTTA 1758

//////////////////////////////////////////////////////////////

                .         .         .
2935 GAGGACACCTGGTACGCGGCTGGGATCTTAAG 2966
     |||||||||||||| ||| ||||||||||||
2709 GAGGACACCTGGTATGCGACTGGGATCTTAAG 2740
```

INPUT FILES

Gap accepts two individual nucleotide sequences or protein sequences as input. The function of Gap depends on whether your input sequence(s) are protein or nucleotide. Programs determine the type of a sequence by the presence of either `Type: N` or `Type: P` on the last line of the text heading just above the sequence. If your sequence(s) are not the correct type, turn to Appendix VI for information on how to change or set the type of a sequence.

RELATED PROGRAMS

When you want an alignment that covers the whole length of both sequences, use Gap. When you are trying to find only the best segment of similarity between two sequences, use BestFit. PileUp creates a multiple sequence alignment of a group of related sequences, aligning the whole length of all sequences. DotPlot displays the entire surface of comparison for a comparison of two sequences. GapShow displays the pattern of differences between two aligned sequences. PlotSimilarity plots the average similarity of two or more aligned sequences at each position in the alignment. Pretty displays alignments of several sequences. LineUp is an editor for editing multiple sequence alignments. CompTable helps generate scoring matrices for peptide comparison.

RESTRICTIONS

Input sequences may not be more than 30,000 symbols long.

ALIGNING LONG SEQUENCES

The program attempts to allocate enough computer memory to align the input sequences. In the worst case, where the two sequences being aligned are unrelated, the allocation is proportional to the product of the lengths of the two input sequences. However, in many cases where the sequences being aligned are more closely related, the computer can determine an optimal alignment using less memory. When memory on your computer is limiting and the program cannot allocate all of the memory it needs to align long sequences, it completes the alignment in whatever memory it can allocate and displays the message `*** Alignment is not guaranteed to be optimal ***`. Because the criteria used in the calculation for guaranteeing an optimal alignment are very stringent, the alignment often may be optimal even if this message is displayed.

If you know roughly where the alignment of interest for long sequences begins, you can run the program with the **-LIMit** command-line parameter. Then set the starting coordinates for each sequence near the point where the alignment of interest begins and set gap shift limits on each sequence. The program then aligns the sequences from your starting point such that the sequences do not get out of phase by more than the gap shift limits you have set. If you started both sequences at base number one and set the gap shift limit for sequence one to 100 and for sequence two to 50, then base 350 in sequence one could not be gapped to any base outside of the range from 300 to 450 on sequence two. These *limited* alignments often require less computer memory than unlimited alignments.

EVALUATING ALIGNMENT SIGNIFICANCE

This program can help you evaluate the significance of the alignment, using a simple statistical method, with the **-RAN**domizations command-line parameter. The second sequence is repeatedly shuffled, maintaining its length and composition, and then realigned to the first sequence. The average alignment score, plus or minus the standard deviation, of all randomized alignments is reported in the output file. You can compare this average *quality* score to the quality score of the actual alignment to help evaluate the significance of the alignment. The number of randomizations can be specified by adding an optional value to **-RAN**domizations; the default is 10.

The score of each randomized alignment is reported to the screen. You can use <Ctrl>C to interrupt the randomizations and output the results from those randomized alignments that have been completed.

By ignoring the statistical properties of biological sequences, this simple Monte Carlo statistical method may give misleading results. Please see Lipman, D.J., Wilbur, W.J., Smith, T.F., and Waterman, M.S. (Nucl. Acids Res. **12**; 215-226 (1984)) for a discussion of the statistical significance of nucleic acid similarities.

CONSIDERATIONS

Other Tools May Be Better Than Gap

Gap is capable of ignoring a region of excellent similarity or similarity between two sequences if it can produce an alignment with equal or better quality in some other way. BestFit is a better tool to search for weak or unknown similarity or similarity that you suspect is not coextensive along the sequences. It is extremely important that you think formally about what Gap does. Using Gap rather than BestFit implies that you want an alignment where neither sequence is truncated.

Gap presents you with one member of the family of best alignments. There may be (and usually are) many members of this family, but no other member has a better *quality*. When two sequences are closely related, Gap is a good way to see the relationship between them; however, a gapped alignment obscures, or can even be confounded by, internal repeats. Graphic matrix analysis is more powerful for seeing internally repeated structures and approximating the frame of best alignment between two sequences that have never been previously compared. (See the Compare and DotPlot programs.)

Scoring Matrices

The modification of scoring matrices is discussed in Appendix VII.

There is considerable evidence that more sensitive nucleic acid alignments may be possible by scoring transitions slightly positive and transversions slightly negative.

Gap chooses default gap creation and extension penalties that are appropriate for the scoring matrix it reads. If you select a different scoring matrix with the **-MATRix** command-line parameter, the program will adjust the default gap penalties accordingly. (See Appendix VII for information about how to set the default gap penalties for any scoring matrix.) You can use **-GAPweight** and **-LENgthweight** to specify alternative gap penalties if you don't want to accept the default values.

CompTable helps you create scoring matrices based on a simplification scheme for amino acid differences. There is a also a short C program that can be modified to help you write a new scoring matrix quickly. The program is called cmpvals.c, and it is located in the public database. You may Fetch and modify cmpvals.c if you are comfortable working with the C programming language.

Forced Pairing

You can get a position in sequence one to pair with some other position in sequence two by choosing a special symbol not used in the rest of the sequences and giving it a very high match value in the scoring matrix. The alphabet of legitimate GCG sequence symbols is defined in Appendix III.

Needleman-Wunsch Versus Sellers

Gap makes an alignment to find the maximum similarity between two sequences by the method of Needleman and Wunsch (J. Mol. Biol. **48**; 443-453 (1970)) that is similar to finding the minimum difference according to the method of Sellers (SIAM J. of Applied Math **26**; 787-793 (1974)). Smith, Waterman, and Fitch (J. Mol. Evol. **18**; 38-46,(1981)) showed that the methods were precisely equivalent when the Needleman and Wunsch gap creation penalty is equal to the Sellers gap creation penalty - 0.5 and when the end gaps for Needleman and Wunsch are penalized in same way as all the other gaps. The command-line parameter **-ENDWeight** allows you to penalize the end gaps introduced by Gap.

Rapid Alignment

When possible, Gap tries to find the optimal alignment very quickly. If this rapid alignment is not unambiguously optimal, Gap automatically realigns the sequences to calculate the optimal alignment. When this occurs, the monitor of alignment progress on your terminal screen (`Aligning...`) is displayed twice for a single alignment.

ALGORITHM

Gap reads a scoring matrix that contains values for every possible GCG symbol match. Gap finds an alignment with the maximum possible quality where the quality of an alignment is equal to the sum of the values of the matches (each match scored with the scoring matrix) less the *gap creation penalty* times the number of internal gaps and less the *gap extension penalty* times the total length of the internal gaps. The alignment found by Gap is, therefore, sensitive to the scoring matrix values and the gap penalties. There is no penalty if either sequence is shifted to the place where the alignment begins unless *end gaps* are penalized by using the command-line parameter **-ENDWeight**.

ALIGNMENT METRICS

BestFit and Gap display four figures of merit for alignments: Quality, Ratio, Identity, and Similarity.

The Quality (described above) is the metric maximized in order to align the sequences. Ratio is the quality divided by the number of bases in the shorter segment. Percent Identity is the percent of the symbols that actually match. Percent Similarity is the percent of the symbols that are similar. Symbols that are across from gaps are ignored. A similarity is scored when the scoring matrix value for a pair of symbols is greater than or equal to the average positive non-identical comparison value in the matrix, the *similarity threshold.* This threshold is also used by the display procedure to decide when to put a ':' (colon) between two aligned symbols. You can change this threshold by specifying the optional values to the **-PAIr** command-line parameter. For instance, the expression **-PAIr=10,5** would set the similarity threshold to 5.

The similarity and identity metrics are not optimized by alignment programs so they should not be used to compare alignments.

PEPTIDE SEQUENCES

If your input sequences are peptide sequences, this program uses a scoring matrix, blosum62.cmp, with comparison values derived from a study of substitutions between amino acid pairs in ungapped block of aligned protein segments as measured by Henikoff and Henikoff (Proc. Natl. Acad. Sci. USA **89**; 10915-10919 (1992)).

COMMAND-LINE SUMMARY

All parameters for this program may be put on the command line. Use the parameter **-CHEck** to see the summary below and to have a chance to add things to the command line before the program executes. In the summary below, the capitalized letters in the parameter names are the letters that you *must* type in order to use the parameter. Square brackets ([and]) enclose parameter values that are optional. For more information, see "Using Program Parameters" in Chapter 3, Using Programs in the User's Guide.

```
Minimal Syntax: % gap [-INfile1=]hpr.seq [-INfile2=]hpf.seq -Default

Prompted Parameters:

-BEGin1=1  -BEGin2=1     beginning of each sequence
-END1=2966 -END2=2740    end of each sequence
-NOREV1    -NOREV2       strand of each sequence
-GAPweight=50            gap creation penalty     (12 is protein default)
-LENgthweight=3          gap extension penalty     (4 is protein default)
[-OUTfile1=]hpr.pair     output file for alignment

Local Data Files: -MATRix=nwsgapdna.cmp scoring matrix for nucleic acids
                  -MATRix=blosum62.cmp  scoring matrix for peptides

Optional Parameters:

-OUTfile2=hpr.gap        new file for sequence 1 with gaps added
-OUTfile3=hpf.gap         "    "    "    "       2  "    "     "
-PENAlizedlength=12      gap extension penalty is applied only to the
                           first 12 positions in a gap
-LIMit1=1 -LIMit2=240    limit the surface of comparison
```

```
-RANdomizations[=10]    determine average score from 10 randomized
                          alignments
-PAIr=x,5,1             thresholds for displaying '|', ':', and '.'
-WIDth=50               the number of sequence symbols per line
-PAGe=60                adds a line with a form feed every 60 lines
-NOBIGGaps              suppresses abbreviation of large gaps with '.'s
-ENDWeight              penalizes end gaps like other gaps
-HIGhroad               makes the top alignment for your parameters
-LOWroad                makes the bottom alignment for your parameters
-NOSUMmary              suppresses the screen summary
```

ACKNOWLEDGEMENTS

Gap and BestFit were originally written for Version 1.0 by Paul Haeberli from a careful reading of the Needleman and Wunsch (J. Mol. Biol. **48**; 443-453 (1970)) and the Smith and Waterman (Adv. Appl. Math. **2**; 482-489 (1981)) papers.

Limited alignments were designed by Paul Haeberli and added to the Package for Version 3.0. They were united into a single program by Philip Delaquess for Version 4.0.

LOCAL DATA FILES

The files described below supply auxiliary data to this program. The program automatically reads them from a public data directory unless you either 1) have a data file with exactly the same name in your current working directory; or 2) name a file on the command line with an expression like **-DATa1=myfile.dat**. For more information see Chapter 4, Using Data Files in the User's Guide.

Local Scoring Matrices

This program reads one or more scoring matrices for the comparison of sequence characters. The program automatically reads the program default scoring matrix file in a public data directory unless you either 1) have a data file with exactly the same name as the program default scoring matrix in your current working directory; or 2) have a data file with exactly the same name as the program default scoring matrix in the directory with the logical name MyData; or 3) name a file on the command line with an expression like **-MATRix=mymatrix.cmp**. If you don't include a directory specification when you name a file on the command line with **-MATRix**, the program searches for the file first in your local directory, then in the directory with the logical name MyData, then in the public data directory with the logical name GenMoreData, and finally in the public data directory with the logical name GenRunData. For more information see "Using a Special Kind of Data File: A Scoring Matrix" in Chapter 4, Using Data Files in the User's Guide.

Gap reads a scoring matrix from your local directory or the public database with the values for every possible match. The file nwsgapdna.cmp (NWS stands for Needleman, Wunsch, and Sellers) has a 10 at every place where the set of bases implied by the alphabetic IUB ambiguity codes (see Appendix III) overlap. All of the other locations have zeros. In the file blosum62.cmp, the scores for pairwise amino acid comparisons range from -4 to +11. You can use the Fetch program to copy, view, and possibly modify these scoring matrix files to suit your own needs.

OPTIONAL PARAMETERS

The parameters listed below can be set from the command line. For more information, see "Using Program Parameters" in Chapter 3, Using Programs in the User's Guide.

-MATRix=mymatrix.cmp

allows you to specify a scoring matrix file name other than the program default. If you don't include a directory specification when you name a file on the command line with **-MATRix**, the program searches for the file first in your local directory, then in the directory with the logical name MyData, then in the public data directory with the logical name GenMoreData, and finally in the public data directory with the logical name GenRunData. For more information see the Local Scoring Matrices topic above.

-PENAlizedlength=12

lets you set the maximum penalty for any gap in the alignment. For instance, if you specify **-PENAlizedlength=12**, then any gap longer than 12 characters is penalized the same as a gap of length 12. Using this parameter, alignments can contain large gaps without incurring large gap extension penalties. This may be useful, for instance, if you are aligning a cDNA sequence with the corresponding genomic DNA sequence containing large introns.

-LIMit1=20 and -LIMit2=20

let you set *gap shift limits* for each sequence. When you already know of a long similarity between two sequences you can "zip" them together using this mode. The beginning coordinates for each sequence must be near the beginning of the alignment you want to see. The alignment continues so that gaps inserted do not require the sequences to get out of step by more than the gap shift limits. You can align very long sequences rapidly. When you set gap shift limits for one or both input sequences, the maximum surface of comparison available to your alignment is 3.5 million. The size of the surface of comparison that your alignment actually requires can be predicted by multiplying the average length of the two sequences by the sum of the two shift limits.

If you add just **-LIMit** to the command line without specifying any value, the program prompts you to enter gap shift limits for each sequence.

-RANdomizations=10

reports the average alignment score and standard deviation from 10 randomized alignments in which the second sequence is repeatedly shuffled, maintaining the length and composition of the original sequence, and then aligned to the first sequence. You can use the optional parameter to set the number of randomized alignment to some number other than 10.

-OUTfile2=seqname1.gap -OUTfile3=seqname2.gap

This program can write three different output files. The first displays the alignment of sequence one with sequence two. The second is a new sequence file for sequence one, possibly expanded by gaps to make it align with sequence two. The third, like the second, is a new sequence file for sequence two, possibly expanded by gaps to make it align with sequence one. The program writes only the first file unless there are output file options on the command line. If there are any output files named on the command line, *only* those output files are written. If you add **-OUT** to the command line without an accompanying file name, then the program will write the second and third output files after prompting you for their names.

Aligned sequences (in sequence files) can be displayed with GapShow. Their similarity can be displayed with PlotSimilarity.

-PAIr=4,2,1

The paired output file from this program displays sequence similarity by printing one of three characters between similar sequence symbols: a pipe character(|), a colon (:), or a period (.). Normally a pipe character is put between symbols that are the same, a colon is put between symbols whose comparison value is greater than or equal to the average positive non-identical comparison value in the scoring matrix, and a period is put between symbols whose comparison value is greater than or equal to 1. You can change these *match display thresholds* from the command line. The three values associated with **-PAIr** are the display thresholds for the pipe character, colon, and period. The match display criterion for a pipe character changes from symbolic identity (the default) to the quantitative threshold you have set in the first parameter. A pipe character will no longer be inserted between identical symbols unless their comparison values are greater than or equal to this threshold. If you still want a pipe character to connect identical symbols, use **x** instead of a number as the first value. (See Appendix VII for more information about scoring matrices.)

-PAGe=60

Printed output from this program may cross from one page to another in an annoying way. Use this parameter to add form feeds to the output file in order to try to keep clusters of related information together. You can set the number of lines per page by supplying a number after **-PAGe**.

-WIDth=50

puts 50 sequence symbols on each line of the output file. You can set the width to anything from 10 to 150 symbols.

-NOBIGGaps

suppresses large gap abbreviations, showing all the sequence characters across from large gaps. Usually, gaps that extend one sequence by more than one complete line of output are abbreviated with three dots arranged in a vertical line.

-ENDWeight

causes the end gaps to be penalized in the same way as all other gaps.

-LOWroad and **-HIGhroad**

The insertion of gaps is arbitrary in many cases, and equally optimal alignments can be generated by inserting gaps differently. When equally optimal alignments are possible, this program can insert the gaps differently if you select either the **-LOWroad** or the **-HIGhroad** parameter. Here are examples for the alignment of GACCAT with GACAT with different parameters.

```
For:        Match = 10          MisMatch = -9
       Gap weight = 10     Length Weight =  0

LowRoad:    1 GACCAT 6
              || |||       Quality = 40
            1 GA.CAT 5
```

```
HighRoad:  1 GACCAT 6
             ||| ||      Quality = 40
           1 GAC.AT 5

For:        Match = 10        MisMatch = 0
      Gap weight = 30    Length Weight = 0

HighRoad:  1 GACCAT 6
             |||           Quality = 30
           1 GACAT. 5

LowRoad:   1 GACCAT 6
               |||         Quality = 30
           1 .GACAT 5
```

Essentially the *low road* shifts all of the arbitrary gaps in sequence two to the left and all of the arbitrary gaps in sequence one to the right. The *high road* does exactly the opposite. When neither *high road* nor *low road* is selected, the program tries not to insert a gap whenever that is possible and uses the high road alternative for all collisions.

-SUMmary

writes a summary of the program's work to the screen when you've used the **-Default** parameter to suppress all program interaction. A summary typically displays at the end of a program run interactively. You can suppress the summary for a program run interactively with **-NOSUMmary**.

You can also use this parameter to cause a summary of the program's work to be written in the log file of a program run in batch.

Printed: November 1, 1996 12:31 (1162)

Exhibit 3

GAP of: Anti-TAC check: 3778 from: 1 to: 87

WPDEF Mouse Anti-TAC Heavy Chain Framework

to: EU check: 3437 from: 1 to: 87

WPDEF Human EU Heavy Chain Framework

Symbol comparison table:
/micro/seqstore/gcg10.0rdb/gcgcore/data/rundata/blosum62.cmp
CompCheck: 6430
BLOSUM62 amino acid substitution matrix.
Reference: Henikoff, S. and Henikoff, J. G. (1992). Amino acid substitution matrices from protein blocks. Proc. Natl. Acad. Sci. USA 89: 10915-10919.

Gap Weight:	8	Average Match:	2.912
Length Weight:	2	Average Mismatch:	-2.003
Quality:	304	Length:	87
Ratio:	3.494	Gaps:	0
Percent Similarity:	72.414	Percent Identity:	66.667

Match display thresholds for the alignment(s):
| = IDENTITY
: = 2
. = 1

Anti-TAC x EU July 1, 1999 21:49 ..

```
                .         .         .         .         .
      1 QVQLQQSGAELAKPGASVKMSCKASGYTFTWVKQRPGQGLEWIGKATLTA 50
        |||| ||||||. |||.|||.|||||| ||.||:| |||||||.|: |:||
      1 QVQLVQSGAEVKKPGSSVKVSCKASGGTFSWVRQAPGQGLEWMGRVTITA 50
                .         .         .
     51 DKSSSTAYMQLSSLTFEDSAVYYCARWGQGTTLTVSS 87
        |.|..||||:|||| ||.| |:||    |  .|||| 
     51 DESTNTAYMELSSLRSEDTAFYFCAGEYNGGLVTVSS 87
```

GAP of: Anti-TAC check: 3778 from: 1 to: 87

WPDEF Mouse Anti-TAC Heavy Chain Framework

to: EU check: 3437 from: 1 to: 87

WPDEF Human EU Heavy Chain Framework

Symbol comparison table:
/micro/seqstore/gcg10.0rdb/gcgcore/data/moredata/pam250.cmp
CompCheck: 5253
PAM250 amino acid substitution matrix.

Gap Weight:	12	Average Match:	2.605
Length Weight:	4	Average Mismatch:	-2.908
Quality:	279	Length:	87
Ratio:	3.207	Gaps:	0
Percent Similarity:	77.011	Percent Identity:	66.667

Match display thresholds for the alignment(s):
| = IDENTITY
: = 2
. = 1

Anti-TAC x EU July 1, 1999 21:44 ..

```
                .         .         .         .         .
       1 QVQLQQSGAELAKPGASVKMSCKASGYTFTWVKQRPGQGLEWIGKATLTA 50
         |||| ||||||: |||.|||:|||||| ||.||:| |||||||:|: |:||
       1 QVQLVQSGAEVKKPGSSVKVSCKASGGTFSWVRQAPGQGLEWMGRVTITA 50
                   .         .         .
      51 DKSSSTAYMQLSSLTFEDSAVYYCARWGQGTTLTVSS 87
         | |..||||:||||  ||.| |:||   .|  :|||| 
      51 DESTNTAYMELSSLRSEDTAFYFCAGEYNGGLVTVSS 87
```

GAP of: Anti-TAC check: 3778 from: 1 to: 87

WPDEF Mouse Anti-TAC Heavy Chain Framework

to: EU check: 3437 from: 1 to: 87

WPDEF Human EU Heavy Chain Framework

Symbol comparison table:
/micro/seqstore/gcg10.0rdb/gcgcore/data/moredata/pep.cmp
CompCheck: 8790
Identity matrix for peptides. This matrix is used as the default for the consensus function for SeqLab protein consensus. All identical matches are scored as 10, and all others (including X-X, and .-.) are scored as 0. Ambiguous peptides (B,Z) match their possible peptides with a score of 10 as well.

Gap Weight:	20	Average Match:	10.000
Length Weight:	1	Average Mismatch:	0.000
Quality:	580	Length:	87
Ratio:	6.667	Gaps:	0
Percent Similarity:	66.667	Percent Identity:	66.667

Match display thresholds for the alignment(s):
| = IDENTITY
: = 10
. = 1

Anti-TAC x EU July 1, 1999 21:47 ..

```
                .         .         .         .         .
       1 QVQLQQSGAELAKPGASVKMSCKASGYTFTWVKQRPGQGLEWIGKATLTA 50
         |||| |||||  ||| ||| |||||| || || | ||||||| |  | ||
       1 QVQLVQSGAEVKKPGSSVKVSCKASGGTFSWVRQAPGQGLEWMGRVTITA 50
                   .         .         .
      51 DKSSSTAYMQLSSLTFEDSAVYYCARWGQGTTLTVSS 87
         | |  |||| ||||  || | | ||    |   ||||
      51 DESTNTAYMELSSLRSEDTAFYFCAGEYNGGLVTVSS 87
```

GAP of: Anti-TAC check: 3778 from: 1 to: 87

WPDEF Mouse Anti-TAC Heavy Chain Framework

to: EU check: 3437 from: 1 to: 87

WPDEF Human EU Heavy Chain Framework

Symbol comparison table:
/micro/seqstore/gcg10.0rdb/gcgcore/data/rundata/blosum62.cmp
CompCheck: 6430
BLOSUM62 amino acid substitution matrix.
Reference: Henikoff, S. and Henikoff, J. G. (1992). Amino acid substitution matrices from protein blocks. Proc. Natl. Acad. Sci. USA 89: 10915-10919.

Gap Weight:	4	Average Match:	2.912
Length Weight:	4	Average Mismatch:	-2.003
Quality:	304	Length:	87
Ratio:	3.494	Gaps:	0
Percent Similarity:	72.414	Percent Identity:	66.667

Match display thresholds for the alignment(s):
| = IDENTITY
: = 2
. = 1

Anti-TAC x EU July 1, 1999 21:50 ..

```
                .         .         .         .         .
      1 QVQLQQSGAELAKPGASVKMSCKASGYTFTWVKQRPGQGLEWIGKATLTA 50
        |||| ||||||. |||.|||.|||||| ||.||:| |||||||.|: |:||
      1 QVQLVQSGAEVKKPGSSVKVSCKASGGTFSWVRQAPGQGLEWMGRVTITA 50
                .         .         .
     51 DKSSSTAYMQLSSLTFEDSAVYYCARWGQGTTLTVSS 87
        |.|..||||:|||| ||.| |:||    |  .|||| 
     51 DESTNTAYMELSSLRSEDTAFYFCAGEYNGGLVTVSS 87
```

GAP of: Anti-TAC check: 3778 from: 1 to: 87

WPDEF Mouse Anti-TAC Heavy Chain Framework

to: EU check: 3437 from: 1 to: 87

WPDEF Human EU Heavy Chain Framework

Symbol comparison table:
/micro/seqstore/gcg10.0rdb/gcgcore/data/moredata/pam250.cmp
CompCheck: 5253
PAM250 amino acid substitution matrix.

Gap Weight:	20	Average Match:	2.605
Length Weight:	20	Average Mismatch:	-2.908
Quality:	279	Length:	87
Ratio:	3.207	Gaps:	0
Percent Similarity:	77.011	Percent Identity:	66.667

Match display thresholds for the alignment(s):
| = IDENTITY
: = 2
. = 1

Anti-TAC x EU July 1, 1999 21:45 ..

```
                .         .         .         .         .
      1 QVQLQQSGAELAKPGASVKMSCKASGYTFTWVKQRPGQGLEWIGKATLTA 50
        |||| ||||| ||| .|||:|||||| ||.||:| |||||||:|: |:||
      1 QVQLVQSGAEVKKPGSSVKVSCKASGGTFSWVRQAPGQGLEWMGRVTITA 50
                .         .         .
     51 DKSSSTAYMQLSSLTFEDSAVYYCARWGQGTTLTVSS 87
        | |..||||:||||  ||.| |:||   .|  :||||
     51 DESTNTAYMELSSLRSEDTAFYFCAGEYNGGLVTVSS 87
```

GAP of: Anti-TAC check: 3778 from: 1 to: 87

WPDEF Mouse Anti-TAC Heavy Chain Framework

to: EU check: 3437 from: 1 to: 87

WPDEF Human EU Heavy Chain Framework

Symbol comparison table:
/micro/seqstore/gcg10.0rdb/gcgcore/data/moredata/pep.cmp
CompCheck: 8790
Identity matrix for peptides. This matrix is used as the default for the consensus function for SeqLab protein consensus. All identical matches are scored as 10, and all others (including X-X, and .-.) are scored as 0. Ambiguous peptides (B,Z) match their possible peptides with a score of 10 as well.

Gap Weight:	10	Average Match:	10.000
Length Weight:	0	Average Mismatch:	0.000
Quality:	580	Length:	87
Ratio:	6.667	Gaps:	0
Percent Similarity:	66.667	Percent Identity:	66.667

Match display thresholds for the alignment(s):
| = IDENTITY
: = 10
. = 1

Anti-TAC x EU July 1, 1999 21:48 ..

```
                .         . ,       .         .         .
  1 QVQLQQSGAELAKPGASVKMSCKASGYTFTWVKQRPGQGLEWIGKATLTA 50
    |||| |||||  ||| ||| |||||| || || | ||||||| |  | ||
  1 QVQLVQSGAEVKKPGSSVKVSCKASGGTFSWVRQAPGQGLEWMGRVTITA 50
                .         .         .
 51 DKSSSTAYMQLSSLTFEDSAVYYCARWGQGTTLTVSS 87
    | |  |||| ||||  || | | ||    |   ||||
 51 DESTNTAYMELSSLRSEDTAFYFCAGEYNGGLVTVSS 87
```